

CLAIMS

WHAT IS CLAIMED IS:

1 1. An apparatus for extracting messages from a digital data stream containing
2 messages, comprising:

3 a message processor that receives the digital data stream and extracts message
4 portions from the digital data stream;

5 a first buffer having a plurality of locations associated with a plurality of channels
6 to store the extracted message portions; and

7 a second buffer having a plurality of locations associated with the plurality of
8 channels for storing state data corresponding to the extracted message portions.

1 2. The apparatus of claim 1, further comprising a central processing unit
2 interface for coupling the apparatus to a central processing unit.

1 3. A device for extracting messages from a data stream, comprising:

2 an input interface that receives packet data in the data stream;

3 a packet identifier filter coupled to the input interface to selectively filter the packet
4 data, the packet identifier filter having a central processing unit (CPU) interface to allow
5 communication between the device and a CPU;

6 a message processor that receives the selectively filtered packet data from the
7 packet identifier filter and extracts message portions from the packet data;

8 a first buffer having a plurality of locations associated with a plurality of channels
9 to store the extracted message portions; and

10 a second buffer having a plurality of locations associated with the plurality of
11 channels for storing state data corresponding to the extracted message portions.

1 4. The device of claim 3, wherein the input interface converts the packet data
2 into parallel packet data.

1 5. The device of claim 4, wherein the parallel packet data is sent to the packet
2 identifier filter with an enable signal to validate byte data in the packet.

1 6. The device of claim 5, wherein the input interface generates at least one
2 clock enable signal to resynchronize the byte data.

1 7. The device of claim 3, wherein the packet identifier filter provides at least
2 one selected from the group consisting of mode control, filtering control, enable control
3 and masking control for each channel in the message processor.

1 8. The device of claim 7, wherein the mode control includes selecting one of a
2 plurality of storage modes, each storage mode corresponding to a buffer size for the first
3 buffer.

1 9. The device of claim 7, wherein the mode control includes selecting one of a
2 capture mode, where the packet data is stored in the first buffer as a full packet without a
3 sync byte, and a message mode, where messages in the packet data are allowed to be
4 processed.

1 10. The device of claim 7, wherein the filtering control includes selecting
2 whether address filtering is turned on or off, and wherein all messages in the packet data
3 are processed when the address filtering is turned on and selected messages in the packet
4 data are processed when the address filtering is turned off.

1 11. The device of claim 3, wherein the filter module has a pipeline delay to
2 allow the packet identifier of an incoming packet to be compared with at least one
3 predetermined packet identifier.

1 12. The device of claim 3, wherein the filter module validates the incoming
2 packet by checking a header in the incoming packet with at least one predetermined
3 condition.

1 13. The device of claim 3, wherein the message processor conducts a first
2 process to find a start of a new message in the packet data and a second process to extract
3 and store the message.

1 14. The device of claim 3, wherein the first buffers are circular buffers.

1 15. The device of claim 14, wherein the first buffer includes 32 available
2 channels each with a 2K buffer for message storage.

1 16. The device of claim 14, wherein the first buffer includes 16 channels with a
2 2K buffer and 4 channels with an 8K buffer for message storage.

1 17. The device of claim 3, wherein the message processor includes:
2 a processor state machine shared between the plurality of channels, wherein the
3 state data from the processor state machine is stored in the second buffer;
4 an address filter control circuit; and
5 a verification circuit that calculates a verification code and compares the calculated
6 verification code with an embedded verification code in the message portion in the packet
7 data.

1 18. The device of claim 17, wherein the message processor further includes an
2 alternative packet capture control that stops message processing for a single channel and
3 captures a single packet for storage in the first buffer.

1 19. The device of claim 17, further comprising a buffer control that controls
2 CPU operation while the at least one of the first and second buffers is being read.

1 20. The device of claim 19, further comprising a message ready interrupt
2 control coupled to the buffer control, wherein the message ready interrupt control generates
3 signals for determining which channels have messages that are ready for processing when
4 the CPU is interrupted based on state data in the second buffer.

1 21. The device of claim 17, further comprising a message error interface for
2 identifying the presence of lost messages.

1 22. The device of claim 21, wherein the message error interface includes a first
2 error circuit that identifies messages lost due to corrupt packets and a second error circuit
3 that identifies messages lost due to first buffer overflow.

1 23. The device of claim 22, wherein the first and second error circuits are
2 provided for each one of said plurality of channels.

1 24. A method for extracting messages from a data stream, comprising:
2 receiving packet data in the data stream;
3 selectively filtering the packet data;
4 extracting at least a portion of a message from the packet data;
5 storing said at least a portion of the message in a first buffer associated with said
6 message processor; and
7 storing state data corresponding with said at least a portion of the message in a
8 second buffer.

1 25. The method of claim 24, further comprising the step of converting the
2 packet data into parallel packet data.

1 26. The method of claim 24, further comprising the step of providing at least
2 one selected from the group consisting of mode control, filtering control, enable control
3 and masking control for each channel in the message processor.

1 27. The method of claim 26, wherein the mode control step includes selecting
2 one of a plurality of storage modes, each storage mode corresponding to a buffer size.

1 28. The method of claim 26, wherein the mode control includes selecting one of
2 a capture mode, where the packet data is stored in the first buffer as a full packet without a
3 sync byte, and a message mode, where messages in the packet data are allowed to be
4 processed.

1 29. The method of claim 26, wherein the filtering control step includes selecting
2 whether address filtering is turned on or off, and wherein the method includes the steps of
3 processing all messages in the packet data when the address filtering is turned on and
4 processing selected messages in the packet data when the address filtering is turned off.

1 30. The method of claim 24, further comprising the step of delaying the data
2 stream to allow the packet identifier of an incoming packet to be compared with at least
3 one programmed packet identifier.

1 31. The method of claim 24, further comprising the step of validating the
2 incoming packet by checking a header in the incoming packet with at least one
3 predetermined condition.

1 32. The method of claim 24, further comprising the steps of:
2 calculating a verification code; and
3 comparing the calculated verification code with an embedded verification code in
4 the message in the packet data.

1 33. The method of claim 32, further including the steps of:
2 selectively stopping message processing for a single channel; and
3 capturing a single packet for storage in the first buffer.

1 34. The method of claim 33, further comprising the step of generating at least
2 one signal for determining which channels have messages that are ready for processing
3 when the CPU is interrupted.

1 35. The method of claim 32, further comprising the step of identifying the
2 presence of lost messages due to at least one of corrupt packets and buffer overflow.